



## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Teori Umum

##### 2.1.1 Pengertian Komputer

Pengertian komputer menurut Siagian (2022:2) “Komputer adalah perangkat elektronik yang menyimpan data, menangani data sesuai dengan instruksi yang diterbitkan secara umum, dan memiliki perangkat elektronik yang dapat menerima dan mengubah data mentah menjadi informasi yang dapat digunakan. Sistem informasi mengumpulkan data, memproses data, dan menghasilkan data”.

Menurut Harmayani et al., (2021:2) komputer terdiri dari 3 elemen utama, antara lain:

1. Perangkat Keras (*Hardware*) yang terdiri dari *Processor*, *RAM*, *Harddisk*, *Motherboard*, dan *CPU*.
2. Perangkat Lunak (*Software*) yaitu sistem operasi dan juga berbagai aplikasi yang dimasukkan ke dalam hardware dan bekerja sesuai perintah dari pengguna.
3. Pengguna Komputer (*Brainware*) yaitu pemakai atau operator komputer.

##### 2.1.2 Pengertian Perangkat Lunak (Software)

Menurut Rahma (2020:2) “*Software* merupakan data elektronik yang disimpan sedemikian rupa oleh komputer, data yang disimpan dapat berupa program atau instruksi yang akan dijalankan oleh perintah, atau catatan-catatan yang diperlukan oleh komputer untuk mencapai tujuannya”.

Sedangkan menurut Habibullah dan Sugiantoro (2023:87) “Pengertian *Software* komputer adalah sekumpulan data elektronik yang disimpan dan diatur oleh komputer, data elektronik yang disimpan oleh komputer itu dapat berupa program atau instruksi yang akan menjalankan suatu perintah”.

##### 2.1.3 Pengertian Internet

Menurut Ni'mah et al., (2019:329) “Pengertian internet adalah layanan jaringan dari komputer yang sifatnya menjangkau internasional dan menggunakan sebuah perangkat jaringan agar bisa terkoneksi ke internet”.



Pada definisi lain, menurut Wilson (2023:22) “Internet ialah, sistem jaringan komputer global yang saling terhubung menggunakan protokol internet (TCP/IP) untuk menghubungkan perangkat komputer di seluruh dunia”.

#### **2.1.4 Pengertian Data**

Menurut Pradana et al., (2022:75) “Data adalah sesuatu yang belum diolah dan belum dapat digunakan sebagai dasar yang kuat dalam mengambil suatu keputusan”.

Sedangkan menurut Kamus Umum Bahasa Indonesia (KUBI) dikutip Baihaqi dan Fansyuri (2022:659) “Data adalah bukti yang ditemukan dari hasil penelitian yang dapat dijadikan dasar kajian atau pendapat”.

### **2.2 Teori Khusus**

#### **2.2.1 Pengertian Unified Modelling Language (UML)**

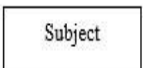
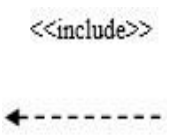
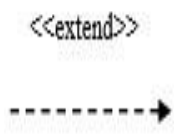
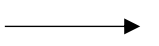
Pakaya et al., (2020:33) menyatakan bahwa *Unified Modeling Language* (UML) merupakan kesatuan struktur dan cara bagi pemodelan desain program berorientasi objek (OOP) serta aplikasinya. UML adalah metodologi untuk mengembangkan sistem OOP dan sekelompok perangkat *tool* untuk mendukung pengembangan sistem tersebut”.

Menurut Putra et al., (2020:35) UML dideskripsikan oleh beberapa diagram, yaitu:

##### **2.2.1.1 Usecase Diagram**

Menurut Syarifuddin & Kurniawan (2020:198-199) “*Usecase Diagram* Merupakan diagram yang bekerja dengan cara mendeskripsikan tipikal interaksi antara *User* (pengguna) sebuah sistem dengan suatu sistem tersendiri melalui sebuah cerita bagaimana sebuah sistem dipakai. *Usecase Diagram* terdiri dari sebuah aktor dan interaksi yang dilakukannya, aktor tersebut dapat berupa manusia, perangkat keras, sistem lain, ataupun berinteraksi sistem”.

Tabel 2.1 Simbol-Simbol *Use Case Diagram*

| Notasi                                                                                          | Nama Elemen                 | Fungsi                                                                                                                                                                |
|-------------------------------------------------------------------------------------------------|-----------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <br>Actor/Role | <i>Actor</i>                | <i>Actor</i> dapat berinteraksi dengan <i>specialization</i> atau <i>superclass association</i> . Actor ditempatkan di luar <i>subject boundary</i> .                 |
|                | <i>Use Case</i>             | Mewakili bagian fungsionalitas sistem dalam <i>system boundary</i> .                                                                                                  |
|                | <i>Subject Boundary</i>     | Menyatakan lingkup dari subjek.                                                                                                                                       |
|                | <i>Include Relationship</i> | Menunjukkan <i>inclusion</i> fungsionalitas sebuah <i>use case</i> dengan <i>use case</i> lainnya. Arah panah dari <i>base use case</i> ke <i>included use case</i> . |
|              | <i>Extend Relationship</i>  | Menunjukkan <i>extension</i> dari sebuah <i>use case</i> untuk menambahkan <i>optional behavior</i> . Arah panah <i>extension use case</i> ke <i>base use case</i> .  |
|              | <i>Association</i>          | Abstraksi dari penghubung antar actor dengan <i>use case</i> .                                                                                                        |
|              | <i>Collaboration</i>        | Interaksi aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemennya.                                             |

Sumber: Putra et al., (2020:35)

### 2.2.1.2 Class Diagram

Menurut Putra et al., (2020:36) “*Class* adalah spesifikasi yang akan menghasilkan objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut) suatu sistem, serta menawarkan layanan untuk memanipulasi keadaan. Kelas memiliki tiga area pokok yaitu nama, atribut dan metode.”

Lanjutan Tabel 2.2 Simbol-Simbol *Class Diagram*

| Notasi | Nama Elemen dan Fungsi                                                                    |
|--------|-------------------------------------------------------------------------------------------|
|        | Kelas pada struktur sistem.                                                               |
|        | Antarmuka. Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.      |
|        | Asosiasi adalah relasi antarmuka, asosiasi biasanya disertai dengan <i>multiplicity</i> . |
|        | Asosiasi berarah artinya makna kelas yang satu digunakan oleh kelas yang lain.            |
|        | Generalisasi adalah relasi antar kelas dengan makna generalisasi-spesialisasi.            |
|        | Agregasi adalah relasi antar kelas dengan makna semua bagian.                             |

Sumber: Putra et al., (2020:36)

### 2.2.1.3 Activity Diagram

Menurut Putra et al., (2020:37) “Menggambarkan rangkaian aliran dari aktivitas dan interaksi beberapa *use case*. Diagram ini sangat mirip dengan *flowchart* karena memodelkan *workflow* dari suatu aktivitas ke aktivitas yang lainnya. Pembuatan *Activity Diagram* pada awal pemodelan proses dapat membantu memahami keseluruhan proses”.

Lanjutan Tabel 2.3 Simbol-Simbol *Activity Diagram*

| Notasi | Nama Elemen                | Fungsi                                                       |
|--------|----------------------------|--------------------------------------------------------------|
|        | <i>Activity</i>            | Merepresentasikan sekumpulan aktivitas.                      |
|        | <i>Control flow</i>        | Menunjukkan rangkaian dari suatu eksekusi.                   |
|        | <i>Initial node</i>        | Pertanda dari suatu awal aktivitas.                          |
|        | <i>Final activity node</i> | Untuk menunjukkan akhir dari suatu aktivitas.                |
|        | <i>Decision node</i>       | Memastikan bahwa alur objek hanya bergerak dalam satu jalur. |
|        | <i>Merge node</i>          | Mengembalikan berbagai <i>decision path</i> menjadi satu.    |
|        | <i>Fork node</i>           | Membagi perilaku menjadi aktivitas yang berjalan bersama.    |
|        | <i>Join node</i>           | Menyatukan aktivitas yang berjalan secara bersamaan.         |

Sumber: Putra et al., (2020:37)

#### 2.2.1.4 Sequence Diagram

Menurut Putra et al., (2020:36) “Menggambarkan interaksi antara sejumlah objek dalam urutan waktu. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara objek dan interaksi antar objek yang terjadi pada titik tertentu dalam eksekusi sistem”.

Lanjutan Tabel 2.4 Simbol-Simbol *Sequence Diagram*

| Notasi                                                                              | Nama Elemen           | Fungsi                                                                                                                                    |
|-------------------------------------------------------------------------------------|-----------------------|-------------------------------------------------------------------------------------------------------------------------------------------|
|    | <i>Entity Class</i>   | Kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.                 |
|    | <i>Boundary Class</i> | Kumpulan kelas menjadi interaksi antar aktor dengan sistem.                                                                               |
|    | <i>Control Class</i>  | Suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas.                                                |
|    | <i>Message</i>        | Simbol mengirim pesan antar <i>class</i> .                                                                                                |
|   | <i>Recursive</i>      | Menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.                                                                        |
|  | <i>Activation</i>     | <i>Activation</i> mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi. |
|  | <i>Lifeline</i>       | Garis terputus dengan objek sepanjang <i>lifeline</i> terdapat <i>activation</i> .                                                        |

Sumber: Putra et al., (2020:36)

### 2.2.1.5 Pengertian Kamus Data

Menurut Sirait dan Seabtian (2019:3) “Kamus data adalah suatu elemen yang terorganisir dengan definisi yang tetap dan sesuai dengan sistem, sehingga *User* dan analis sistem mempunyai pengertian yang sama *input*, *output*, dan *componen data store*. Kamus data dibuat pada tahap analisis sistem dan digunakan baik pada tahap analisis maupun pada tahap perancangan sistem. Pada tahap analisis, Kamus data digunakan sebagai alat komunikasi antara analisis sistem dengan pemakai sistem tentang data yang mengalir disistem yaitu tentang data yang masuk ke sistem dan tentang informasi yang dibutuhkan oleh pemakai sistem.”.

**Lanjutan Tabel 2.5** Simbol-Simbol pada Kamus Data (*Data Dictionary*)

| No | Simbol | Keterangan                     |
|----|--------|--------------------------------|
| 1. | =      | Disusun atau terdiri dari      |
| 2. | +      | Dan                            |
| 3. | [1]    | Baik...atau...                 |
| 4. | {}     | n kali diulang/bernilai banyak |
| 5. | ()     | Data opsional                  |
| 6. | *...*  | Batas Komentar                 |

Sumber: (Sirait dan Seabtian 2019:4)

## 2.3 Teori Judul

### 2.3.1 Pengertian Aplikasi

Menurut Wulandari et al., (2021:12) “Aplikasi adalah suatu komputer, intruksi dan pernyataan yang disusun sedemikian rupa sehingga komputer dapat memproses input menjadi output. Aplikasi adalah suatu kelompok file yang bertujuan untuk melakukan aktivitas tertentu yang saling berkaitan”

Menurut Huda & Priyatna, (2019:82) “Aplikasi dapat diartikan sebagai suatu program berbentuk perangkat lunak yang berjalan pada suatu sistem tertentu yang berguna untuk membantu berbagai kegiatan yang dilakukan oleh manusia.”

### 2.3.2 Pengertian Sponsorship

Menurut Chusnaini dan Rasyid, (2023:51) “Sponsorship adalah elemen paling penting dalam komunikasi pemasaran untuk setiap jenis perusahaan dan organisasi di antara jenis-jenis komunikasi pemasaran lainnya.”

Menurut Utomo et al., (2022:7) “Sponsorship menjadi suatu upaya dalam suatu perusahaan untuk dapat dipromosikan dan atau dipasarkan baik secara langsung atau tidak langsung melalui upaya sponsorship yang mereka lakukan melalui pihak ketiga atau yang disponsori.”



### 2.3.3 Tanggung Jawab Sosial Lingkungan (TJSL)

TJSL PT Pusri Palembang bertujuan untuk memberikan kemanfaatan bagi pembangunan ekonomi, pembangunan sosial, pembangunan lingkungan serta pembangunan hukum dan tata kelola bagi perusahaan, memberikan kontribusi pada penciptaan nilai tambah bagi perusahaan dengan prinsip yang terintegrasi, terarah dan terukur dampaknya dan membina usaha mikro dan usaha kecil agar lebih tangguh dan mandiri, serta masyarakat sekitar perusahaan.

### 2.3.4 Website

Menurut Syabania dan Rosmawani (2021:46) “*Website* merupakan sebuah halaman berisi informasi yang dapat dilihat jika komputer terkoneksi dengan internet. Dengan adanya *website*, semua orang di dunia bisa mendapatkan dan mengelola informasi dengan berbagai sumber yang tersedia di internet”.

Pada definisi lain, menurut Noviana (2022:113) “*Website* adalah Halaman web yang saling berhubungan yang berisi kumpulan informasi berupa teks, gambar, animasi, audio dan video bisa diakses melalui jalur koneksi internet yang dibuat untuk personal, organisasi dan perusahaan.

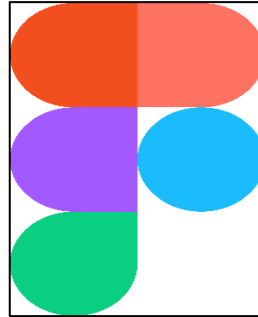
### 2.3.5 Pengertian Aplikasi Pengolahan Pengajuan Proposal Sponsorship Pada Departemen Tanggung Jawab Sosial dan Lingkungan PT Pupuk Sriwidjaja Palembang Berbasis Website

Aplikasi ini adalah sebuah aplikasi berbasis web yang dirancang untuk memudahkan proses pengajuan proposal sponsorship kepada Departemen Tanggung Jawab Sosial dan Lingkungan (TJSL) PT Pupuk Sriwidjaja Palembang. Dengan menggunakan aplikasi ini, masyarakat dapat mengajukan proposal sponsorship secara *online* tanpa perlu datang langsung ke kantor. Aplikasi ini juga membantu departemen dalam menerima, memverifikasi, dan menilai proposal dengan lebih efisien.



## 2.4 Teori Program

### 2.4.1 Pengertian Figma



**Gambar 2.1** Logo Figma

*Sumber: figma.com*

Menurut Surianto et al. 2023:59) “Figma adalah aplikasi desain grafis berbasis web yang memungkinkan pengguna untuk membuat desain antarmuka pengguna (UI) dan pengalaman pengguna (UX) yang interaktif. Aplikasi ini sangat populer di kalangan desainer UI/UX karena kemampuannya untuk memungkinkan kolaborasi tim secara *real-time* dan menyediakan fitur-fitur yang memudahkan proses desain. Selain itu, Figma juga memiliki keunggulan dalam hal kemudahan penggunaan dan aksesibilitas karena dapat diakses melalui browser web tanpa perlu mengunduh aplikasi terlebih dahulu”.

### 2.4.2 Pengertian HTML



**Gambar 2.2** Gambar HTML

*Sumber: <https://en.wikipedia.org/wiki/HTML>*

Menurut Syabania dan Rosmawani (2021:46) “*HTML* adalah bahasa untuk menyebarkan informasi pada *web*. Ketika merancang *HTML*, ide ini diambil dari



*Standard Generalized Markup Language (SGML)*. *HTTP* adalah protokol komunikasi *stateless* yang berbasiskan pada *TCP* yang awalnya digunakan untuk mengambil kembali *file-file* *HTML* dari server web ketika dirancang pada tahun 1991”.

Pada definisi lain, menurut Noviana (2022:113) “*Hypertext Markup Language (HTML)* adalah script pemrograman yang mengatur bagaimana kita menyajikan informasi di dunia internet dan bagaimana informasi itu membawa kita melompat dari satu tempat ke tempat lainnya”.

### 2.4.3 Pengertian CSS



**Gambar 2.3** Logo CSS

Sumber: <https://en.wikipedia.org/wiki/CSS>

Menurut Noviana (2022:113) “*Cascading Style Sheet (CSS)* merupakan salah satu kode pemrograman yang bertujuan untuk menghias dan mengatur gaya tampilan atau *layout* halaman web agar lebih elegan dan menarik. *CSS* adalah sebuah dokumen yang berdiri sendiri dan dapat dimasukkan dalam kode *HTML* atau sekedar menjadi rujukan oleh *HTML* dalam pendefinisian *style*. Ada banyak hal yang dapat dilakukan menggunakan *CSS* dibandingkan dengan bahasa pemrograman inti seperti *HTML* dan *PHP*”.



#### 2.4.4 Pengertian PHP



**Gambar 2.4** Logo PHP

*Sumber: <https://en.wikipedia.org/wiki/PHP>*

Menurut Syabania dan Rosmawani (2021:46) “PHP adalah *script* bersifat *server-side* yang ditambahkan ke dalam HTML. PHP sendiri merupakan singkatan dari *Personal Home Page Tools*. *Script* ini akan membuat suatu aplikasi dapat diintegrasikan ke dalam HTML sehingga suatu halaman *website* tidak lagi bersifat statis, namun menjadi dinamis”.

Menurut Noviana (2022:114) “PHP adalah bahasa pemrograman *script server-side* yang didesain untuk pengembangan web. Selain itu, PHP juga bisa digunakan sebagai bahasa pemrograman umum. PHP diciptakan oleh Rasmus Lerdorf pertama kali tahun 1994. Saat ini PHP adalah singkatan dari PHP: *Hypertext Preprocessor*, sebuah kepanjangan rekursif, yakni permainan kata dimana kepanjangannya terdiri dari singkatan itu sendiri: PHP: *Hypertext Preprocessor*”.



#### 2.4.5 Pengertian XAMPP



**Gambar 2.5** Logo XAMPP

*Sumber: <https://en.wikipedia.org/wiki/XAMPP>*

Menurut Syabania dan Rosmawani (2021:46) “*Cross-platform (X), Apache, MariaDB/MySQL, PHP, Perl (XAMPP)* adalah perangkat lunak bebas, yang mendukung banyak sistem operasi. Fungsinya adalah sebagai server yang berdiri sendiri (localhost), yang terdiri atas program *Apache* HTTP Server, *MySQL* database, dan penerjemah bahasa yang dirilis dengan bahasa pemrograman *PHP* dan *Perl*”.

#### 2.4.6 Pengertian MySql



**Gambar 2.6** Logo *My Structured Query Language* (MySQL)

*Sumber: <https://en.wikipedia.org/wiki/MySQL>*

Menurut Syabania dan Rosmawani (2021:46) “*MySQL* adalah *database* yang menghubungkan *script* *PHP* menggunakan perintah *query*. Fungsi dari halaman ini adalah sebagai pengendali *database* *MySQL* sehingga pengguna *MySQL* tidak perlu repot untuk menggunakan perintah-perintah *SQL*”